MANUAL DE USO
Y SISTEMA OPERATIVO

PECOS
SISTEMAS

MANUAL DE USO Y SISTEMA OPERATIVO

INDICE GENERAL

CAPITULO 1 – INTRODUCCION AL SISTEMA OPERATIVO

CONCEPTOS DE COMPUTACION 1
EL COMPUTADOR: UNA HERRAMIENTA DE ORGANIZACION 1
EL PRIMER NIVEL: EL HARDWARE 2
Entrada/Salida 2
Las Instrucciones 3
EL SEGUNDO NIVEL: EL PROGRAMA 4
Programas 4
Hardware vs. Software 5
La Memoria de Trabajo o RAM 6
EL TERCER NIVEL: EL SISTEMA OPERATIVO 6
El Sistema de Archivos 8
Programas y Programación 9
Lenguajes de programación 9
Intérpretes y Compiladores 10
EL CUARTO NIVEL: LOS PROGRAMAS DE APLICACION 12

CAPITULO 2 – EL SISTEMA OPERATIVO

PARTES QUE CONSTITUYEN EL DOS 13
El BIOS 14
El BDOS 14
El CCP 16
¿DE DONDE VIENE EL SISTEMA OPERATIVO? 17

CAPITULO 3 – EL DOS, EL COP Y LOS COMANDOS

LOS COMANDOS 18
LAS REFERENCIAS A ARCHIVOS 19
LAS REFERENCIAS DE ARCHIVO MULTIPLES 20
EL SIGNO DE INTERROGACION - UN COMODIN 20
CONMUTACION DE DISCOS 22

CAPITULO 4 – EL TECLADO

LAS TECLAS ESPECIALES 23
La tecla de ESCAPE o ESC 24
La tecla BACKSPACE o BS 24

La tecla de TABULADO o TAB 24
La tecla de CONTROL o CTRL 24
La tecla RETURN 25
Las teclas LOCK y SHIFT 25
La tecla HOME 26
Las teclas de FLECHA ARRIBA, ABAJO, IZQUIERDA Y DERECHA 26
MODALIDAD DEL TECLADO 27
LA ENE 27

CAPITULO 5

EDICION DE LINEA BAJO EL CCP 28
CONTROL DE IMPRESION 28
SUSPENSION DE LA SALIDA A CONSOLA 29
CONTROL-C o REBOOT 29
CONMUTACION DE PANTALLA EN MODO COLOR 30
VUELCO DE PANTALLA GRAFICA A IMPRESOR 30
BORRADO DE PANTALLA 30

CAPITULO 6 – LOS COMANDOS INTERNOS

Comando DIR 31
Comando ERA 32
Comando REM 33
Comando TYPE 34

CAPITULO 7 – COMANDOS TRANSITORIOS

Comando FORMAT 36
Comando GENSIS 38
Comando COPIA 39
Comando BASIC 41
Comando GBASIC 41
Comando STAT 42
Comando COL40 43
Comando COL40B 43
Comando COLOR 43
Comando SUBMIT 44

CAPITULO 8 – MODOS DE FUNCIONAMIENTO

MODO COLOR 46
MODO 40 COLUMNAS 47

CAPITULO 1

INTRODUCCION AL SISTEMA OPERATIVO

(COMPATIBLE CON CP/M)

CONCEPTOS DE COMPUTACION

Este capítulo es para gente que ha tenido escaso o ningún contacto previo con computadoras. Físicamente una computadora es para la mayoría de las personas un objeto familiar. El teclado, la pantalla, la impresora son elementos frecuentemente vistos en la vida cotidiana, sin embargo el concepto se presenta más difuso: por ejemplo, ¿para qué sirve?, ¿por dónde se comienza?, ¿cómo comprender el uso del equipo como un todo?

Existen ciertos conceptos que son comunes a todas las computadoras; debemos familiarizarnos con éstos y con las palabras que los designan antes de poder comprender los manuales de uso del equipo. Trataremos de hacer aquí una introducción a estos temas para luego avanzar al uso de su computadora **PECOS.**

EL COMPUTADOR: UNA HERRAMIENTA DE ORGANIZACION

Obviamente comenzaremos por definir el término COMPUTADOR. Si usted no conoce estos equipos esta palabra puede inducirnos a pensar que se trata básicamente de una herramienta de CALCULO o COMPUTACION. Si bien es cierto que ésta es una función importante de las computadoras, no es de ninguna manera su función fundamental. Las computadoras pasan, en general, la mayor parte de su tiempo buscando, ordenando, organizando y por último mostrando datos. Una computadora es en esencia una herramienta que ordena elementos según un patrón y luego almacena o muestra estos elementos o datos una vez ordenados.

EL PRIMER NIVEL: EL HARDWARE

Denominamos HARDWARE (del inglés 'ferretería') a la parte electrónica tangible de una computadora. En su nivel fundamental estos equipos ordenan patrones de impulsos eléctricos. El patrón de ordenamiento está dado por las interconexiones electrónicas internas, que hoy en día están mayormente dentro de los circuitos integrados (pequeñas obleas de silicio encapsuladas en plástico con conexiones metálicas), o sea los célebres CHIPS'.

ENTRADA/SALIDA

Un chip de silicio es inútil si no tiene manera de comunicarse con el resto del mundo. Antes que una computadora pueda pasar de ser un ordenador de impulsos eléctricos a ser un asistente útil o siquiera un controlador, deberá acoplarse al entorno que la rodea.

En la jerga, estos acoples se denominan dispositivos I/O (del inglés Input/Output). Casi cualquier fenómeno físico puede ser representado por impulsos eléctricos. Por ejemplo, un magnetófono convierte el sonido en impulsos eléctricos, éstos en magnéticos, y los almacena en la capa de óxido de una cinta o cassette.

Una computadora manipula impulsos eléctricos, así que si encontramos una forma de convertir un efecto físico a algún tipo de impulsos eléctricos compatibles, entonces la computadora podrá recibirlos y manipularlos. Este fenómeno lo vemos siempre desde el lado de la computadora y por lo tanto a esto denominamos INPUT o entrada.

El magnetófono es capaz de reproducir el sonido almacenado como impulsos magnéticos en la cinta convirtiéndolos de vuelta en un efecto físico (el sonido). Por analogía, si podemos conectar al chip algún dispositivo que reconvierta los impulsos eléctricos ordenados por la computadora en un fenómeno físico, entonces la computadora podrá hacer que algo suceda en el mundo exterior. Esto se denomina OUTPUT o salida.

Considerando que prácticamente cualquier cosa se puede convertir a impulsos eléctricos y que virtualmente todo es controlable eléctricamente, se ve que una computadora se puede insertar en casi cualquier proceso.

LAS INSTRUCCIONES

Las maneras en las que un chip de computadora puede ordenar las cosas está limitada por el circuito de interconexiones interno, y son pocas y sencillas. En realidad, ninguna computadora puede hacer muchas cosas, pero lo que hace lo hace a muy alta velocidad.

Entre las secuencias definidas para un chip como los que estamos describiendo puede estar, por ejemplo: tomar una señal de una parte del chip, combinarla con otra de tal otra parte y guardar el resultado en tal otro lugar. Mediante la especificación de las maneras en que las señales son combinadas y/o almacenadas, los diseñadores del chip crean secuencias que hacen operaciones aritméticas simples. Algunas secuencias copian un dato de un lugar a otro, otras reciben señales desde líneas de entrada, otras las emiten por líneas de salida, aun otras comparan impulsos y emiten una señal que indica cuál de las dos antes mencionadas es la mayor.

Cada operación básica incluida dentro del circuito de un chip de computadora se denomina INSTRUCCION y el conjunto de instrucciones posibles se llama SET de INSTRUCCIONES. Cada instrucción es ejecutada invocándola por su CODIGO DE OPERACION (que es en esencia sencillamente un número).

El número 20, recibido como entrada puede querer decir 'SUMAR DOS NUMEROS', mientras que 36 puede significar 'ACEPTAR UN DATO DE TAL DISPOSITIVO'.

El set de instrucciones de un computador es llamado frecuentemente su LENGUAJE DE MAQUINA, aunque en rigor más que de un lenguaje se trata de un vocabulario.

EL SEGUNDO NIVEL: EL PROGRAMA

Hasta ahora hemos tenido una visión microscópica o interna de una computadora. Podemos ahora alejarnos un poco de los chips, cables y sus interconexiones.

Este segundo nivel aparece cuando proveemos al microprocesador de un PROGRAMA, o sea una lista ordenada de instrucciones pertenecientes a su lenguaje de máquina.

Podemos ahora observar cómo la computadora o PROCESADOR recorre esta lista o programa, ejecutando una tras otra las operaciones correspondientes, según su set de instrucciones, a muy alta velocidad.

El programa, visto así, no es más que un libreto de palabras del vocabulario del procesador y sería de escasa utilidad, salvo por el hecho de que algunas de las instrucciones del set permiten modificar el orden en el que se ejecuta el libreto, basado en el resultado de operaciones ejecutadas previamente.

Por ejemplo, la lista, programa (o libreto si usted quiere) puede decir algo así como: "si la suma de dos números es cero, ejecute estas instrucciones, en caso contrario ejecute estas otras", o bien algo como: "repita la siguiente serie de instrucciones hasta que se cumpla tal o cual condición".

PROGRAMAS

Hemos dicho que a la lista de instrucciones a ejecutar la llamamos programa. Ninguna instrucción de por sí es demasiado poderosa, pero un par de miles de instrucciones bien elegidas, ejecutadas a alta velocidad y en el orden adecuado pueden, sí, hacer tareas de alta complejidad.

El poder de la computadora deviene de su capacidad de ejecutar un programa, y porque se puede hacer que un programa modifique su propio orden de ejecución bifurcando éste basado en una elección, o bien ejecutando una repetición (LOOP o lazo) un número de veces. Sin un programa, la máquina no tiene sentido y es un conjunto inútil de partes electrónicas. Con un programa se puede lograr que maneje sus dispositivos I/O de cualquier manera que la persona que escribe el programa desee. Si se cambia el programa, la máquina puede hacer otra tarea distinta.

HARDWARE VS. SOFTWARE

Podemos ahora volver a mirar la definición de dos términos importantes: HARDWARE y SOFTWARE. El hardware se compone de los chips, algunos dispositivos eléctricos y mecánicos y los cables que los interconectan, junto con uno o más dispositivos de I/O (por ejemplo, un teclado y una pantalla).

El único propósito del HARDWARE, la única razón de su existencia, es hacer posible la ejecución de un programa. Llamamos SOFTWARE al conjunto de programas que se usan con una computadora.

La distinción entre hardware y software es de vital importancia. Podemos ilustrarla y fijar conceptos con la siguiente metáfora: un programa es a una computadora como un disco es a un fonógrafo. El software es a un sistema de cómputos como una colección de discos es a un sistema de alta fidelidad.

Así como un tocadiscos no tiene utilidad sin discos para tocar, una computadora es inútil sin un programa para ejecutar.

Los programas se escriben en cierta manera como partituras musicales, o sea que un programa es similar a una partitura, es la copia de las instrucciones para ejecutar algo, no una copia de la ejecución en sí.

LA MEMORIA DE TRABAJO O RAM

El programa ejecutado por un chip de computadora no está contenido dentro del chip en sí. Viene de afuera, de a una instrucción por vez. Pero de afuera, ¿adónde? Viene de un elemento llamado MEMORIA (RAM - random access memory o memoria de acceso al azar). La memoria se compone de un puñado de circuitos integrados, cada uno de los cuales contiene unos miles de unidades de datos o códigos de instrucción o letras o números.

Cuando el chip de computadora está listo para ejecutar una instrucción, le "dice" a la RAM: necesito el contenido de la posición de memoria número tal o cual. La RAM responde con el contenido del número requerido en la posición o celda requerida y la computadora interpreta este contenido como una instrucción de acuerdo con su vocabulario (juego de instrucciones [instruction set], como ya vimos).

el número que designa una posición dentro de la memoria se denomina DIRECCION (address, en inglés) y la acción de emitir un address o dirección se denomina DIRECCIONAMIENTO.

EL TERCER NIVEL: EL SISTEMA OPERATIVO

En un primer nivel vimos a la computadora como ordenador y procesador de señales eléctricas. En un segundo nivel vemos a la máquina ejecutando un programa compuesto de pasos elementales. Para convertirse en una herramienta realmente útil y versátil, una computadora necesita muchos programas y la capacidad de moverse de uno a otro fácil y rápidamente.

Por empezar necesita un conjunto de programas que le permitan llevar a cabo tareas sencillas pero fundamentales, como almacenar un grupo de datos para luego poder recuperarlos, o cargar un nuevo programa cuando llega el momento de cambiar de una tarea a otra. El conjunto de programas que ejecutan estas tareas de entrecasa o proveen estos SERVICIOS (como se dice en la jerga) se denomina SISTEMA OPERATIVO.

EL SISTEMA OPERATIVO ES UN CONJUNTO DE PROGRAMAS QUE APLICAN LA POTENCIA DE LA COMPUTADORA A LA SUPERVISION DE LAS TAREAS QUE ESTA EJECUTA.

Por esta razón se denomina a veces al sistema operativo como MONITOR o SUPERVISOR.

El Sistema Operativo, en adelante S.O. (del inglés Operating System) contiene diversos tipos de programas. Algunos se encargan de manejar los dispositivos de I/O de la máquina, otros tienen la misión de cargar determinados programas en la memoria para ser ejecutados y aun otros llevan cuenta de conjuntos de datos llamados ARCHIVOS (files, en inglés) además de permitir crearlos, leerlos, guardarlos y copiarlos de un dispositivo I/O a otro.

Una vez que hemos llegado a este tercer nivel, la máquina propiamente dicha deja de interesarnos, excepto en su función de ejecutar los programas del S.O. Nuestra atención se desplaza hacia los programas. De este nivel en adelante deja de importarnos el hardware que usamos, ya que todo lo que vemos son los programas. Para los usuarios la computadora en sí no es más que un medio de ejecutar programas. Salvo en lo que respecta a su costo o confiabilidad, el hardware pierde totalmente su relevancia.

Ilustrando lo recién dicho con un ejemplo de la vida cotidiana podemos decir lo siguiente: cuando recurro a una ferretería para adquirir un tornillo, basta con que se lo pida al ferretero (sistema operativo), éste se ocupa de obtener el tornillo de la gaveta en que está guardado y de entregármelo. Yo sólo tuve que solicitar un tornillo, el "sistema operativo" me relevó del detalle de saber precisamente dónde éste estaba guardado, prestándome el "servicio" requerido sin la necesidad de que yo, el "usuario", tuviera que interactuar con la estantería (hardware) de la ferretería.

EL SISTEMA DE ARCHIVOS

La computadora se usa para almacenar, manipular y mostrar datos y estos datos se almacenan en unidades llamadas archivos, que se identifican mediante un NOMBRE. Uno de los componentes fundamentales de todo S.O. es el de proveer un servicio llamado SISTEMA DE ARCHIVOS (file system, en inglés) a través del cual podamos crear colecciones de datos, asignarles un nombre y almacenarlos en DISCOS o CINTAS o DISPOSITIVOS MAGNETICOS. El sistema de archivos incluye programas para la creación, borrado y asignación o cambio de nombres de los mencionados archivos. Otros programas nos permitirán copiar archivos de un dispositivo a otro (de un disco a otro, de un disco a la impresora, etc.).

Los discos son el medio de almacenaje permanente de una computadora. Los archivos guardados (SALVADOS, en la jerga) en disco permanecen sin alteraciones hasta tanto el usuario los borre o modifique.

Los archivos no sólo guardan datos. También hay archivos de programas. Un programa debe ser leído de un archivo de disco y ubicado en RAM para ser ejecutado (CORRIDO es el término habitualmente empleado).

Correr o ejecutar un programa significa aplicar la computadora a la ejecución de las instrucciones del programa.

Otra función importante del S.O. la cumple un programa llamado PROCESADOR DE COMANDOS DE CONSOLA, cuya función es la de correr, o iniciar la ejecución de un programa, a requerimiento del usuario, trayéndolo de un archivo de disco a la memoria de trabajo, cuando éste es invocado por su nombre.

PROGRAMAS Y PROGRAMACION

La palabra programa significa, según el diccionario, "un esquema de tareas a ser ejecutadas, o un plan preconcebido de procedimiento". Antes dijimos que un conjunto de instrucciones de máquina era un programa. Este es nuestro "plan de procedimiento" expresado en lenguaje de máquina. Cuando está lista es cargada en memoria, la computadora puede ejecutar las instrucciones a alta velocidad resolviendo así la tarea planeada o programada.

Los programas son compuestos o escritos por personas, usando los servicios que ofrece el S.O. El acto de componer un plan para ser ejecutado por una computadora se denomina PROGRAMACION y la persona que genera el programa se llama PROGRAMADOR.

El plan delineado en un programa expresa los pasos necesarios para la solución de un problema. Tales soluciones paso a paso se denominan ALGORITMOS.

Dado un problema, el trabajo del programador consiste en encontrar el algoritmo que implementa su solución y expresar éste en forma de programa.

LENGUAJES DE PROGRAMACION

Gracias al Sistema Operativo, el programador no se ve obligado a escribir el programa como una larga secuencia de números, sino que puede expresar el algoritmo en un LENGUAJE DE PROGRAMACION. Existen muchos de éstos. Un lenguaje de programación es realmente un lenguaje con vocabulario, gramática y reglas de sintaxis. Cada uno de estos elementos es más restringido que el correspondiente elemento de, por ejemplo, el idioma castellano. Es decir, sería prácticamente imposible mantener una conversación en un lenguaje de programación, ya que éste no es lo suficientemente flexible, ni su vocabulario es suficientemente general ni extenso como para mantener un coloquio verbal.

Un lenguaje de programación es tan sólo un lenguaje artificial, diseñado para facilitar la expresión de soluciones algorítmicas a ciertas clases de problemas.

Algunos problemas son fácilmente expresables en cierto lenguaje de programación y difíciles en otro. Por ejemplo, la solución a problemas científicos se expresa eficientemente en el lenguaje FORTRAN y difícilmente en COBOL, mientras que este ultimo es más adecuado que el primero para expresar soluciones a problemas comerciales.

Existe una clase de lenguajes, denominados LENGUAJES DE PROPOSITOS GENERALES, que se adaptan a la implementación de soluciones algorítmicas a una amplia gama de problemas. El lenguaje BASIC es un ejemplo de lenguaje de propósito general.

También existen lenguajes muy especializados, que permiten una muy elevada eficiencia de expresión a una franja muy estrecha de problemas.

INTERPRETES Y COMPILADORES

Un programa se escribe en el lenguaje elegido y se ingresa a la computadora tipeándolo mediante letras, números y símbolos de puntuación.

El texto del programa, en general, se almacena en un archivo. La computadora, como ya sabemos, no puede ejecutar palabras, sólo puede ejecutar el código de las instrucciones de su vocabulario. Por ende, antes de poder ejecutar un programa escrito en un lenguaje superior, éste debe ser traducido. Cada oración o SENTENCIA del programa en lenguaje superior deberá ser traducido a una secuencia de tal vez docenas de instrucciones en lenguaje de máquina que producen el efecto especificado por la sentencia original.

Este proceso de traducción es llevado a cabo por un PROGRAMA TRADUCTOR.
Existen en general dos grandes familias de programas traductores: los INTERPRETES y los COMPILADORES.

Un INTERPRETE es un programa que toma cada sentencia del lenguaje superior y ejecuta las instrucciones de lenguaje de máquina que esa sentencia representa. El intérprete traduce y ejecuta el programa al mismo tiempo. Tanto el intérprete como el programa que está siendo interpretado deben estar simultáneamente presentes en memoria, de manera que los intérpretes dejan en general un espacio de memoria disponible para datos y programa relativamente limitado. Como la traducción se hace al mismo tiempo que la ejecución, el tiempo que la máquina tarda en traducir una sentencia debe adicionarse al tiempo de ejecución de ésta, lo cual limita la velocidad de ejecución. La ventaja del intérprete reside en la facilidad de DEPURACION (debugging, en inglés) del programa. La depuración es el proceso de búsqueda y subsaneamiento de errores lógicos o de sintaxis en el programa que implementa el algoritmo deseado. Al estar en memoria el PROGRAMA FUENTE (el programa en lenguaje superior), sin traducir, es sencillo introducir cambios y observar sus efectos.

Otros traductores son COMPILADORES. Estos programas hacen la traducción del texto completo del programa fuente una única vez. La lista de instrucciones en lenguaje de máquina así generada se almacena en un archivo en disco. Esta versión traducida se llama PROGRAMA OBJETO. Este programa objeto puede ser cargado en memoria por el sistema operativo y ser ejecutado. En términos generales, un programa compilado corre mucho más velozmente y deja más espacio disponible de memoria que un programa interpretado. No obstante, es más difícil y laborioso depurarlo, además de requerir un mayor dominio del lenguaje y del S.O. para escribirlo.

BASIC y Pascal son dos lenguajes de propósitos generales bastante comunes. En general, BASIC es un lenguaje intérprete y Pascal un compilador. Esto no quita que existan versiones compiladas de BASIC e interpretadas de Pascal.

EL CUARTO NIVEL: LOS PROGRAMAS DE APLICACION

El conjunto de software del sistema operativo, el sistema de archivos, los programas utilitarios y traductores de lenguajes (también llamados simplemente 'lenguajes'), hacen posible que el programador construya APLICACIONES.

Un programa de aplicación, APLICATIVO o simplemente "aplicación" es todo aquel programa dedicado exclusivamente a resolver los problemas de la gente, en lugar de resolver los problemas internos o gerenciales de una computadora (estos últimos, que por supuesto incluyen el OS, traductores, etc., son a veces llamados PROGRAMAS DE SISTEMA).

La denominación "aplicación" cubre prácticamente cualquier cosa que se puede hacer con una computadora más allá de programarla: juegos, programas de enseñanza, procesamiento de textos, simulaciones de ingeniería, contabilidad, costos, sueldos son todos aplicaciones.

A excepción del tiempo empleado en escribir programas, la mayor parte del tiempo de una computadora se emplea en ejecutar aplicaciones.

Las aplicaciones constituyen el cuarto nivel desde el cual podemos ver la computadora. La elegancia microscópica de los chips, los ingeniosos mecanismos de los impresores y unidades de disco y la sofisticación y complejidad del sistema operativo existen meramente para permitir la programación y ejecución de aplicaciones.

Sólo las aplicaciones hacen trabajo útil para ayudar al hombre en su tarea, de modo que sólo las aplicaciones pueden justificar el costo de una computadora.

CAPITULO 2

EL SISTEMA OPERATIVO

En la primera parte de este texto hemos pasado revista a la terminología que emplearemos de ahora en más. Sabemos qué es una computadora, cuáles son las partes que la componen y hemos visto someramente la interacción entre el hardware y el software. También hemos visto las distintas clases de software distinguiendo entre el software de sistema y los lenguajes, utiiitarios y aplicaciones.

Esta parte del manual está destinada a familiarizarnos con el sistema operativo, pieza fundamental del software de una computadora, ya que, como vimos, las aplicaciones, lenguajes, etc., que usaremos verán a la computadora a través del sistema operativo. Dicho de otra manera, el sistema operativo provee servicios al resto del software, permitiendo que éstos se ocupen de sus tareas específicas sin necesidad de ocuparse de los detalles íntimos de los componentes de lo que hemos llamado el primer nivel.

La computadora PECOS utiliza un poderoso sistema operativo denominado CP/M (iniciales de Control Program for Microprocessors). Este sistema operativo reside en disco magnético, de donde es cargado en memoria de trabajo, y entre los servicios que brinda está fundamentalmente el sistema de archivos. Por esta razón el CP/M pertenece a una familia de S.O. denominada en inglés "disk operating system" o DOS. En adelante utilizaremos esta sigla para nombrar al sistema operativo.

PARTES QUE CONSTITUYEN EL DOS

Ya sabemos que el DOS es un programa, o más precisamente un conjunto de programas. El DOS que utiliza la PECOS consta básicamente de tres partes denominadas CCP, BDOS y BIOS.

EL BIOS

BIOS significa "Basic Input - Output system". Así como en el capítulo previo vimos que el microprocesador con su set de instrucciones está rodeado de capa tras capa de software, desde el lenguaje de máquina hasta las aplicaciones, pasando por el DOS; dentro del DOS mismo existen subcapas. El BIOS es la capa interna del DOS, la más íntimamente vinculada al hardware. Así como el DOS está diseñado para ocultar los detalles del hardware, simplificando su uso, la función del BIOS es la de ocultar los detalles de los PERIFERICOS (dispositivos de entrada/salida) frente al resto del DOS. Aquí nuevamente el propósito es que las capas o estratos superiores no necesiten preocuparse de detalles físicos de los dispositivos de E/S. Por ejemplo, el subsistema de manejo de archivos se comunica con la unidad de disco a través de los servicios del BIOS, y no precisa preocuparse por la cantidad de pistas, tamaño, etc., de los discos empleados. Resumiendo, diremos que el BIOS es la interfase entre los dispositivos de entrada/salida de la computadora y el resto del DOS.

EL BDOS

El BDOS constituye el núceo central del DOS. BDOS significa "basic disk operating system". Como sabemos es técnicamente software de sistemas. En rigor, es una colección de programas diseñada para proveer, al software de niveles superiores, los servicios de los que tanto hemos hablado. Son fundamentalmente tres categorías de servicios los provistos: manejo de consola, manejo de impresor y manejo de archivos.

La consola es el dispositivo de E/S a través del cual el operador se comunica con la computadora. La entrada o ingreso de datos se produce a través del teclado y la salida a través de la pantalla, que puede ser un monitor, televisor monocromo o color. El manejo de la consola a través del sistema operativo está implementado por programas que permiten recibir un texto del teclado y/o mandar un mensaje a la pantalla.

El impresor es un elemento (puede no estar presente en algunas configuraciones) que nos permite obtener salida de datos sobre papel (esto se denomina HARD COPY, en inglés). El BDOS ofrece programas para enviar mensajes al impresor.

El sistema de archivos es la parte más compleja del BDOS. Todos los servicios que presta este conjunto de programas está dedicado al mantenimiento de archivos en discos magnéticos. Antes de pasar a ver en detalle los servicios disponibles es conveniente entender los conceptos de disco, archivo y registro. Usando una analogía con elementos comunes de oficina, podemos visualizar al disco como una gaveta de un armario donde se almacenan carpetas que, a su vez, contienen papeles (facturas, cartas, recibos, etc.). La acción de elegir un cajón del armario y abrirlo es análoga a elegir un disco e introducirlo (MONTARLO, en la jerga) en la unidad de disco. En general, el cajón contiene un cierto número de carpetas, posiblemente separadas por temas. Estas carpetas tienen su paralelo con los archivos. Un cajón (disco) contiene múltiples carpetas (archivos). A su vez cada carpeta contiene una cierta cantidad de papeles, cuyo paralelo en una computadora son los registros. Resumiendo, un disco contiene una cierta cantidad de archivos, cada uno de los cuales contiene uno o más registros. Los registros, por último, contienen los datos, o sea la información. El BDOS contiene programas que permiten crear y eliminar archivos. También existen programas para escribir, o sea agregar y/o modificar datos de los archivos, así como leerlos, o sea recuperar la información almacenada en ellos. Una función fundamental del sistema de archivos es el mantenimiento de lo que se denomina la ESTRUCTURA DE DIRECTORIO. Esta estructura es ni más ni menos que lo que su nombre implica. El DOS mantiene, en una zona especialmente designada del disco, un directorio o índice de los archivos ahí contenidos. Este índice es similar al de un libro, es decir, figuran en el directorio los nombres de todos los archivos existentes en el disco, así como los datos numéricos que indican la ubicación física de los registros de cada archivo en el disco. Cuando se desea acceder a un archivo a través del BDOS, se lo hace especificando su nombre.

El BDOS se encarga de buscar ese nombre en el directorio y de ubicar los datos donde sea que éstos se encuentren en toda la extensión del disco. Surge aquí la necesidad de otra función del sistema de archivos del BDOS: la de modificar el nombre de un archivo. Cuando un archivo se crea, aparece su nombre en el directorio.

Cuando se lo borra, se elimina del directorio el nombre correspondiente. Cuando se lee un registro de un archivo, el BDOS busca en el directorio la ubicación física del registro solicitado, y cuando se agrega un dato a un archivo, el BDOS actualiza el directorio agregando, bajo el nombre correspondiente al archivo, un número que representa la ubicación del nuevo dato.

EL CCP

El CCP (Console Command Processor) es un programa de características muy especiales. Si bien es parte del sistema operativo, también tiene los atributos de un programa de aplicación. El CCP es el programa que la computadora ejecuta cuando no está ejecutando ninguna tarea especificada por el operador. Su función principal es la de recibir órdenes del usuario a través de la consola a efectos de poner en funcionamiento (correr) el programa solicitado por éste. También el CCP implementa ciertos comandos (más adelante los veremos) llamados comandos internos o residentes que permiten ciertas funciones, como por ejemplo la visualización del directorio del disco. A efectos prácticos, estos comandos internos son como programas utilitarios, con la diferencia que al estar incorporados dentro del DOS, no precisan ser traídos de disco para su ejecución, lo cual hace sumamente veloz el acceso a ellos. Por último, diremos que el CCP cuando está actuando indica al usuario que se encuentra listo para recibir una orden mediante un símbolo en la pantalla. Este símbolo se denomina PROMPT. En el DOS de la PECOS, el prompt del CCP consiste en una letra, seguida del símbolo "mayor".

¿DE DONDE VIENE EL SISTEMA OPERATIVO?

Cuando se enciende la computadora, la memoria normal de trabajo no contiene información útil alguna. Para que se pueda empezar a trabajar con el equipo se precisa instalar (traer) el sistema operativo a memoria y poner en funcionamiento el CCP.

La letra indica la unidad de disco que está en uso, en general para la PECOS, la letra será "A" y el signo ">" indica que a continuación se puede escribir un comando. Cada vez que en el margen izquierdo de la pantalla veamos el Pompt:

A>

sabemos que el CCP está dispuesto para recibir nuestras órdenes.

El sistema operativo, en forma de programa objeto, reside en las dos primeras pistas de un disco magnético. Estas pistas se denominan PISTAS DE SISTEMA. Estas pistas están destinadas y reservadas al DOS. Todo disco que contenga efectivamente el DOS en esta ubicación se denomina DISCO DE SISTEMA. Si el DOS no se encontrase grabado en las pistas de sistema, el disco igual sirve para almacenar archivos, pero no para arrancar la máquina. Veremos que existe un programa utilitario que nos permite grabar el DOS en las pistas de sistema de un disco que no lo tuviera.

Cuando encendemos la computadora y/o presionamos el pulsador de RESET, un pequeño programa almacenado en una memoria especial, distinta de la memoria normal de trabajo, busca la primera parte del sistema operativo en el disco de sistema, lo instala en memoria y hace que comience a funcionar. Este primer segmento del DOS se ocupa de cargar en memoria el CCP, el BDOS y el BIOS, transfiriendo luego el control al CCP. El CCP pone el prompt en pantalla y el equipo está listo para su uso. Todo este proceso de carga en dos etapas que hemos descripto se denomina BOOTSTRAP, y la acción de iniciar este proceso por parte del operador se denomina RESETEAR o BOOTEAR la máquina. Con seguridad que la Real Academia Española no aceptaría ninguno de estos neologismos, pero nuestra intención es que el usuario se exprese y entienda solventemente en el ámbito de la computación reconociendo que la tecnología avanza mucho más rápido que la evolución de la lengua.

CAPITULO 3

EL DOS, EL CCP Y LOS COMANDOS

Hemos visto que luego de resetear la máquina, un programa almacenado en una memoria especial, se encarga de traer del disco de sistema a memoria principal el DOS compuesto del CCP, el BDOS y el BIOS. También vimos que el CCP es un pequeño programa cuya misión es poner en pantalla un prompt, avisando al usuario que está dispuesto a recibir comandos.

LOS COMANDOS

Denominamos COMANDO a toda palabra ingresada por el usuario a continuación del prompt del CCP, seguida de un caracter de RETURN o retorno de carro. Existen fundamentalmente dos tipos de comandos: los COMANDOS INTERNOS y los COMANDOS TRANSITORIOS.

Son comandos internos aquellos que están incorporados dentro de la estructura del CCP, y que por lo tanto están siempre disponibles en memoria principal y no precisan ser traídos de disco para ser ejecutados.

El CCP tiene implementados los siguientes comandos, cuyas funciones describiremos luego en detalle:

- DIR
- ERA
- TYPE
- REN

Los comandos externos son aquellos que el CCP se ocupa de traer de disco para su ejecución. Para esto deben estar presentes en el directorio del disco montado en la disketera.

Los comandos pueden ser ingresados en letras mayúsculas o minúsculas, indistintamente.

Los comandos, en general se componen del nombre del comando, seguido de una frase denominada COLA DE COMANDO (puede no ser necesaria, de acuerdo al comando en cuestión). La cola de comando es en general una REFERENCIA A ARCHIVO, o file reference, en inglés.

LAS REFERENCIAS A ARCHIVO

Una referencia a archivo identifica por nombre a un archivo o grupo de archivos sobre un disco en particular, conectado al DOS. Estas referencias a archivo pueden ser referencias UNIVOCAS (RUA - referencia unívoca de archivo) o MULTIPLES (RMA - referencia múltiple de archivo). Una referencia unívoca identifica, como su nombre lo indica, a un único archivo, mientras que la referencia múltiple es satisfecha por un número de archivos diferentes.

Las referencias a archivos constan de tres partes: el designador de disco, el NOMBRE PRIMARIO Y EL NOMBRE SECUNDARIO, también llamado EXTENSION. La forma genérica de una referencia de archivo es:

d:pppppppp.eee

donde d: representa el designador de disco; pppppppp: el nombre primario, y eee: la extensión, separada del nombre primario por un caracter "." (punto).

El designador de disco consta de una letra seguida de ":" (dos puntos). La letra identifica la unidad de disco en la que esperamos encontrar el archivo designado.

En la PECOS, son válidas las letras A y B si se trata de un equipo con dos disketeras y "A" solamente en caso de poseer una única disketera. En todos los casos se podrá omitir el designador de disco, entonces el CCP interpretará que el archivo se refiere a la unidad de diskette que está activa en ese momento. (Recordemos que la letra que precede al signo ">" en el prompt indica precisamente cuál es el disco que se halla activo en un momento dado).

El nombre primario, que genéricamente hemos designado pppppppp, consta de entre uno y ocho caracteres. En una referencia de archivo unívoca, el primer caracter no puede ser un blanco y tampoco se pueden usar en ninguna posición los caracteres < > . , ; = ? ¡ ¿ (mayor, menor, punto, coma, punto y coma, igual, interrogación, corchete abierto y corchete cerrado).

El nombre secundario o extensión es también opcional. De estar presente debe estar separado del nombre primario por un punto y debe constar de uno a tres caracteres, y están vedados los mismos que para el nombre principal. La extensión juega el papel de "apellido" en la referencia a archivo, ya que en general se lo usa para identificar "familias" de archivos.

LAS REFERENCIAS DE ARCHIVOS MULTIPLES

Las referencias de archivo múltiples difieren de las vistas hasta ahora en la posibilidad de incluir entre los caracteres del nombre primario y de la extensión el caracter "?" (interrogación), vedado como ya vimos en las referencias unívocas.

Dicho de otra manera, la diferencia de formato entre una referencia unívoca y una múltiple es la presencia de uno o varios signos de interrogación esparcidos a lo largo del nombre primario y/o de la extensión de la referencia a archivo.

EL SIGNO DE INTERROGACION - UN COMODIN

Cuando se procesa una referencia de archivo, el proceso subyacente es el de buscar en el directorio del disco aquel archivo cuyo nombre primario y extensión (si la hay) concuerdan exactamente, posición por posición con la referencia a archivo deseada.

La función del signo de interrogación es la de comodín, o sea que a medida que se va comparando la referencia de archivo múltiple contra cada uno de los nombres contenidos en el directorio, para una posición dada, desde la primera letra del nombre primario hasta la última de la extensión, se da por satisfecha la concordancia si suceden una de dos cosas:

- El caracter de la posición explorada de la referencia de archivo coincide exactamente con el caracter de la misma posición en el directorio del disco.
- El caracter de la posición explorada es un signo de interrogación.

Una referencia unívoca como:

SUELDOS.DAT

sólo es satisfecha en una búsqueda por un archivo cuyo nombre primario es SUELDOS con la extensión DAT y por ningún otro; en cambio la referencia múltiple

SUELDOS?.D?T

es satisfecha por cualquiera de los siguientes archivos (si estuvieran en el disco designado)

SUELDOS1.DAT

SUELDOS2.DAT

SUELDOSX.DIT

pero no por

SUELDOS1.IAT

Obsérvese que existen casos especiales de gran importancia. El primero y más genérico es la referencia a archivo:

????????.???

Esta referencia múltiple engloba a TODOS los archivos contenidos en el directorio de un disco, mientras que las siguientes referencias:

????????.XYZ

ABCDEFGH.???

designan, respectivamente, a todos los archivos cuya extensión es XYZ (la "familia" XYZ) para el primer ejemplo, y a todos los archivos de nombre ABCDEFGH, sea cual fuere su extensión para el segundo ejemplo.

Para comodidad del operador, se permite representar los ocho signos de interrogación que forman un nombre primario múltiple o los tres signos de interrogación que componen una extensión múltiple por un único asterisco "*".

De esta manera la referencia totalmente incluyente:

????????.???

se convierte en:

.

y los dos ejemplos antes mencionados de nombre primario múltiple y extensión múltiple quedan como sigue:

*.XYZ

ABCDEFGH.*

Nótese que ambas notaciones son válidas, tanto los múltiples signos de interrogación como los asteriscos. La diferencia radica exclusivamente en la comodidad.

CONMUTACION DE DISCOS

Hemos dicho que de no figurar explícitamente un designador de disco en una referencia a archivo, ésta se toma como referida al disco asumido, o sea aquel cuya letra precede al ">" del prompt. El CCP tiene un mecanismo para cambiar el disco asumido (siempre que la computadora tenga al menos dos unidades de disco).

Por ejemplo, para pasar del "A" como disco asumido a "B", basta con ingresar el comando B: seguido de return. Se verá cómo inmediatamente el prompt cambia de

A>

a

B>

De ahí en más, toda referencia unívoca o múltiple que no contenga explícitamente un designador de disco, se referirá al directorio del disco asumido, en este caso el "B".

CAPITULO 4

EL TECLADO

Es conveniente estudiar detenidamente el teclado de la PECOS antes de pasar a estudiar el comportamiento de los comandos internos y transitorios, ya que toda nuestra interacción con la computadora se hará a través de este elemento.

El teclado es muy similar al de una máquina de escribir común. No obstante existen teclas que cumplen funciones que son específicas de una computadora y por ende no tienen paralelo en una máquina de oficina.

La mayor extensión del teclado, su parte central, está ocupada por un teclado alfanumérico. Las teclas de la primera hilera (la más alejada del operador), excluidas la primera y la última; las teclas de la segunda hilera, excluida la primera; las de la tercera hilera, excluida la primera y la última; las de la cuarta hilera, excluidas las primeras dos y la última, y la barra de espaciado cumplen en la PECOS las funciones acostumbradas. Es decir cada una genera el caracter que figura dibujado en el lugar correspondiente. Las teclas que tienen un solo símbolo, generan el caracter en mayúscula o minúscula de acuerdo al modo operativo y al estado de las teclas de SHIF y LOCK que veremos enseguida. Aquellas teclas que tienen dos símbolos, generan uno u otro también de acuerdo al estado de SHIFT y LOCK.

LAS TECLAS ESPECIALES

Veremos ahora la función de las teclas que quedaron excluidas a la explicación del párrafo anterior.

Seguiremos el orden de la primera hilera a la última.

LA TECLA DE ESCAPE O ESC

La primera tecla de la primera hilera genera el caracter "ESCAPE". Este es un caracter especial que significa que lo que sigue debe ser interpretado de una manera particular. Básicamente el significado de ESC seguido de uno o más caracteres (esto se denomina SECUENCIA DE ESCAPE) depende de la aplicación que está corriendo y se usa, en general, para enviar comandos al sistema. En el apéndice de comandos de gráficos y color se ve un amplio repertorio de secuencias de escape y sus significados. A nivel del CCP, o sea durante el ingreso de comandos esta tecla no se utiliza.

LA TECLA BACKSPACE O BS

La tecla BS es la última de la primera hilera. BACKSPACE significa "retroceso" y, en general, el caracter emitido por esta tecla se utiliza para retroceder en el ingreso de texto. Esta tecla es de útil uso durante el ingreso de comandos, como veremos más adelante.

LA TECLA DE TABULADO O TAB

La primera tecla de la segunda hilera emite al oprimirse el caracter TAB. La acción de digitar TAB depende del programa que esté corriendo. En general causa un tabulado a la próxima columna que sea múltiplo de 8.

LA TECLA DE CONTROL O CTRL

Esta es la primera tecla de la tercera hilera. Es una tecla muy especial, ya que no genera ningún caracter. Bajo control de cualquier programa (desde el CCP hasta cualquier aplicación) oprimir la tecla de CTRL aisladamente no produce acción alguna. Esta tecla es un modificador de los códigos generados por las demás teclas. Dicho de otra manera, la tecla de CTRL se usa SIEMPRE en conjunto con otra tecla, en general alfabética. De ahora en más, cuando digamos, por ejemplo, oprima CONTROL C, nos estaremos refiriendo a la acción de oprimir la tecla CTRL y la tecla C, SIMULTANEAMENTE.

Más adelante veremos que hay una cantidad de códigos de control (CTRL digitado junto con otra tecla) que cumplen funciones especiales.

En adelante representaremos la combinación de CTRL y una tecla alfabética, por ejemplo CTRL y "C" como ^C. Cada vez que el texto indique oprima ^C, interpretaremos que se debe pulsar las teclas CTRL y "C" simultáneamente.

LA TECLA RETURN

Se trata de la última tecla de la tercera hilera. Esta tecla emite un caracter especial que generalmente significa: fin del mensaje o comando. Todos los comandos dados al CCP se terminan con esta tecla. Los datos ingresados durante un programa de aplicación también se terminan en RETURN.

LAS TECLAS LOCK Y SHIFT

La primera y segunda tecla de la cuarta hilera, pulsadas individualmente, producen el cambio a mayúsculas. Al igual que la tecla de CTRL, estas teclas no emiten código pulsadas individualmente, sino que modifican el código de teclas oprimidas simultáneamente con ellas. Oprimir la tecla "A" genera una letra a minúscula, pero pulsando SHIFT y "A" simultáneamente obtenemos la A mayúscula. Aquellas teclas que tienen dos símbolos, generan el de más abajo, salvo cuando simultáneamente se oprime SHIFT, en cuyo caso se genera el símbolo superior. Por ejemplo, la tecla del número 2 genera este número cuando actúa individualmente y genera la comilla cuando se la activa junto con SHIFT.

Oprimiendo SIMULTANEAMENTE la tecla CTRL y LOCK se obtiene la función de traba de mayúsculas. En este caso, las teclas alfabéticas generan siempre la mayúscula, SALVO QUE ADEMAS SE OPRIMA SHIFT, en cuyo caso se genera la minúscula correspondiente.

EN EL ESTADO DE LOCK SE INVIERTEEL FUNCIONAMIENTO DE SHIFT: LOS CARACTERES SIEMPRE SON MAYUSCULA, SALVO CUANDO SON DIGITADOS JUNTO CON SHIFT. ESTO ES EXACTAMENTE LA INVERSA DE LO QUE SUCEDE CUANDO NO SE ESTA EN LOCK.

El estado de LOCK NO FUNCIONA IGUAL A LA TRABA DE MAYUSCULAS DE UNA MAQUINA DE ESCRIBIR. En esta última, la traba actúa tanto sobre las letras como los símbolos. El estado de LOCK de una computadora sólo actúa sobre las teclas alfabéticas. Los símbolos y teclas de dos símbolos siguen respondiendo al comando de la tecla SHIFT. Sea cual fuere el estado de LOCK, oprimir el "2" genera el número 2 y oprimir SHIFT y 2 simultáneamente genera la comilla, tal como está dibujada encima del "2" en la tecla correspondiente.

Habiendo entrado en estado de LOCK, oprimiendo como se mencionó antes CTRL y LOCK simultáneamente, la única manera de revertir el estado de LOCK es digitar nuevamente CTRL y LOCK.

LA TECLA HOME

La última tecla de la cuarta hilera es HOME. Esta tecla genera un código cuya interpretación depende de la aplicación. Es básicamente una tecla de control de posición del cursor (el cursor es el puntero que indica dónde va a escribirse en pantalla el próximo caracter), siendo su función la de retornar el cursor a la fila 1 columna 1, o sea el vértice superior izquierdo de la pantalla. Recalcamos que la función de esta tecla depende del programa. En el caso específico del CCP, no tiene aplicación.

LAS TECLAS DE FLECHA ARRIBA, ABAJO, IZQUIERDA Y DERECHA

Estas cuatro teclas, ubicadas en pares a izquierda y derecha de la barra espaciadora, son similares a HOME en tanto y en cuanto su acción depende del programa y su utilización primaria es la de mover el cursor en la dirección de la flecha respectiva. No tienen aplicación bajo el CCP, salvo la FLECHA IZQUIERDA, cuya función es idéntica a la tecla de BACKSPACE PUDIENDO utilizarse indistintamente una u otra.

MODALIDAD DEL TECLADO

Existiendo en la PECOS diversos modos operativos, por ejemplo MODO COLOR y MODO 40 COLUMNAS, algunas teclas tienen diferencias operativas de acuerdo al modo en que se trabaje.

Fundamentalmente, en modo COLOR, si bien todo lo dicho respecto a SHIFT, LOCK y a mayúsculas y minúsculas es válido, la representación en pantalla será siempre en mayúsculas. Remarcamos que sólo la representación visual es siempre mayúscula, ya que internamente sigue distinguiéndose entre mayúsculas y minúsculas, fenómeno comprobable si emitimos un texto por impresora.

También cabe hacer notar que en modo COLOR, los símbolos ~ ñ ` ¨ } (tilde, barra partida, acento grave y llave abierta y cerrada) no son activos. En modo 40 COLUMNAS estas teclas cumplen con sus funciones indicadas.

LA EÑE

El código interno usado por las computadoras es el ASCII (American Standard Code for Information Interchange) no contempla la existencia de la letra EÑE. No obstante se ha standarizado el uso de la tecla "Ñ" (barra invertida) para representar la EÑE. Es por esto que cada vez que se digite "Ñ" aparecerá en pantalla la letra EÑE.

CAPITULO 5

EDICION DE LINEA BAJO EL CCP

Cuando el CCP pone en pantalla el prompt y está dispuesto a recibir una línea de comando, incluye un pequeño editor interno que permite corregir eventuales errores de tipeo del comando antes de dar el RETURN.

Esta posibilidad de edición está dada por las teclas BS o FLECHA IZQUIERDA (indistintamente) y ^X (CTRL y X digitadas simultáneamente). La tecla BS borra el último caracter digitado y ^X borra la totalidad de la línea de comando.

Digitando ^R el CCP repite en una nueva línea la porción de línea de comando ingresada hasta ese momento, permitiendo seguir tipeando.

CONTROL DE IMPRESION

Si durante el ingreso de un comando (y aun dentro de algunos programas de aplicación) se digita ^P, el sistema manda todo aquello que aparece en pantalla simultáneamente al impresor.

NOTA: Si el impresor no estuviera encendido, o estando encendido no estuviera ON-LINE (dispuesto a imprimir), y eventualmente en caso de no haber un impresor conectado a la PECOS, digitar ^P puede hacer que el sistema "se cuelgue". Esto significa que se detiene la operación de la computadora, ya que ésta manda cada caracter tanto a la pantalla como a la impresora, y si ésta no está en condiciones de recibir el caracter enviado, el sistema esperará hasta que se subsane esta condición.

Una vez digitado ^P, o sea cuando se está en "modo eco", basta con volver a digitar ^P para volver a la condición inicial.

SUSPENSION DE LA SALIDA A CONSOLA

En general, cuando el sistema está mandando datos a la pantalla (y eventualmente ^P por medio del impresor) puede suceder que lo haga a una velocidad demasiado elevada para permitir una lectura de los datos. En estos casos, digitando ^S se suspende la emisión de datos hasta tanto se digite otra tecla cualquiera.

Es recomendable utilizar ^S para continuar la emisión de datos una vez suspendidos, aunque como antes se dijo, cualquier tecla (salvo CTRL, LOCK y SHIFT) cumplirá el mismo efecto.

CONTROL-C O REBOOT

Hemos dicho que la acción de cargar al sistema operativo se la suele llamar BOOTEAR el equipo. Existen ciertas circunstancias en las que se quiere recomendar el funcionamiento del DOS sin necesariamente volver a cargarlo de disco. A esta operación se la denomina REBOOT o WARM-START (arranque en caliente).

Desde el CCP se puede iniciar un REBOOT digitando ^C.

Es fundamental hacer un ^C cada vez que se cambia el diskette que está montado. Esto se debe a que cuando el DOS accede a un diskette por primera vez, hace una lectura total del directorio, a los efectos de almacenar en memoria el MAPA DE ALOCACION. Esto no es más que tomar conocimiento de los lugares disponibles y ocupados del diskette. Si cambiamos el diskette y tratáramos de escribir sobre el nuevo diskette basados en el mapa de alocación del que estaba antes montado, los resultados pueden ser catastróficos, ya que espacios disponibles en un diskette pueden estar ocupados en otro. El DOS antes de escribir en un disco, verifica que éste se corresponda con el mapa de alocación que tiene en memoria, y si detecta discrepancias emite un mensaje de error (BDOS ERR ON A: R/O). Para evitar esto, al cambiar de diskette debemos ingresar un ^C al sistema. Esto hace que se actualice el mapa de alocación de acuerdo al nuevo disco.

En general, en la mayoría de los programas utilitarios y de aplicación ^C significa abandonar la tarea y devolver control al CCP.

CONMUTACION DE PANTALLA EN MODO COLOR

Cuando se opera en modo COLOR, existen en realidad dos pantallas, una sola de ellas está visible (ver apéndice de modo gráfico).

La digitación de ^W exhibe la pantalla alfanumérica y ^E exhibe la pantalla gráfica. Fuera del modo COLOR, ^W y ^E no actúan.

VUELCO DE PANTALLA GRAFICA A IMPRESOR

En modo COLOR, es posible imprimir el diseño gráfico de la pantalla homónima digitando ^E para pasar a exhibir el gráfico (si éste no estuviese en pantalla) y luego oprimiendo ^V.

Valen para este caso las mismas prevenciones respecto a la posibilidad de colgar el sistema que las enumeradas para la secuencia ^P.

NOTA: Si se oprime ^V cuando se encuentra exhibida la pantalla alfanumérica, la impresora procederá a imprimir el generador de caracteres alfanumérico en lugar del gráfico deseado. Si el gráfico no está en pantalla, se lo debe traer a ella con ^E antes de digitar ^V.

BORRADO DE PANTALLA

Si bien no existe una secuencia del CCP que permita borrar la pantalla, a veces es conveniente poder hacerlo para evitar confusión con datos anteriormente exhibidos. Esto se puede lograr de una manera poco ortodoxa, pero efectiva, digitando ^Z y RETURN.

CAPITULO 6

LOS COMANDOS INTERNOS

Recordemos que las referencias a archivos podían ser de dos tipos: Referencias Unívocas (RUA) y Referencias Múltiples (RMA). Con esta nomenclatura podremos explicar la estructura de los comandos internos.

COMANDO DIR

El comando DIR tiene por objeto exhibir en la pantalla (y también en la impresora digitando ^P) parte o la totalidad del directorio de un disco.

El formato del comando es:

DIR rma (return)

Serán listados en pantalla todos aquellos nombres de archivo que estando contenidos en el directorio del diskette especificado por la rma (o por el disco asumido si la rma no incluyera referencia de disco) que satisfagan la rma. También es válido usar:

DIR rua (return)

ya que una rua no es más que un caso especial de rma.

De esta manera el comando

DIR SUELDOS?.DAT (return)

exhibirá todos los archivos del disco asumido cuya extensión sea DAT y cuyo nombre sea SUELDOS seguido de un caracter cualquiera.

En forma similar el comando

DIR B:MOVIM.* (return)

mostrará en pantalla todos los archivos del disco B (suponiendo que esté instalado) cuyo nombre sea MOVIM con cualquier extensión o nombre secundario.

Por último, el comando

DIR *. (return)

pondrá en pantalla el nombre de la totalidad de los archivos contenidos en el directorio del disco asumido (ya que no hay referencia explícita de disco) sin importar su nombre primario o secundario.

NOTA: En el caso del comando DIR, y sólo en este caso,

DIR *. (return)

es equivalente a:

DIR (return)

En caso de que no exista en el diskette designado ningún archivo que satisfaga la RUA, aparece en pantalla el mensaje: NO FILE.

COMANDO ERA

El comando ERA sirve para borrar uno o varios archivos del directorio del disco.

El formato del comando es:

ERA rma (return)

cuya acción es la de borrar del directorio del disco designado explícitamente por la RMA o implícitamente por el disco asumido, todos aquellos archivos que satisfagan la RMA. Estos archivos quedarán totalmente bloqueados y no podrá accederse a ellos para leer o escribir, ni aparecerán más en los listados de directorio. También el espacio previamente ocupado por el o los archivos borrados pasará a estar disponible para nuevos datos.

Como ejemplo,

ERA *.BAS (return)

borrará del disco asumido todos aquellos archivos cuyo nombre secundario sea BAS.

En el caso especial del comando

ERA *.* (return)

cuyo efecto es el borrado de la totalidad del contenido del disco, el DOS emite el mensaje:

ALL FILES (Y/N)?

Debiendo el usuario confirmar, digitando la Y (de yes) que efectivamente es su intención borrar la totalidad del disco.

Si no existen en el disco archivos que satisfagan la RMA el sistema emite el mensaje: NO FILE.

COMANDO REN

Este comando permite cambiar el nombre (primario y secundario) de un archivo.

Su formato es

REN rua2=rua1 (return)

El efecto es que el archivo que satisface a RUA1 pasa a llamarse como RUA2. En caso que ya exista en el disco un archivo de nombre RUA1, aparece el mensaje:

FILE EXISTS

En caso de no existir archivo que satisfaga a RUA1 obtendremos el mensaje:

NO FILE

A modo de ejemplo supongamos que tenemos un archivo llamado CAJA.001 y deseamos renombrarlo a CAJA.002. El comando correcto es:

REN CAJA.002=CAJA.001 (return)

Este comando sólo admite referencias unívocas de archivo. No es posible renombrar más de un archivo por cada aplicación del comando.

COMANDO TYPE

El comando TYPE permite visualizar en pantalla (y eventualmente imprimir vía ^P) el contenido de cualquier archivo cuyo contenido sean caracteres alfanuméricos.

El formato del comando es:

TYPE rua (return)

El objetivo es mostrar por pantalla el archivo cuyo nombre satisface la RUA. En caso de no existir en el directorio ningún archivo en estas condiciones, aparece el mensaje NO FILE.

Si por error se pide un TYPE de un archivo que no contiene caracteres alfanuméricos (por ej.: un comando transitorio) el efecto es impredecible, probablemente llenando la pantalla de símbolos ininteligibles.

Durante la ejecución del TYPE se puede detener la salida de datos a pantalla digitando ^S ya descripto. Esto permite visualizar textos que son más largos que la pantalla, adecuando la velocidad de escritura a comodidad del lector.

CAPITULO 7

COMANDOS TRANSITORIOS

Existen, aparte de los comandos internos del CCP (DIR, ERA, TYPE y REN), comandos transitorios o externos. El nombre de transitorios es porque no se encuentran permanentemente en memoria listos para ser ejecutados, sino que existen como archivos en un diskette. Todos los archivos de comando del DOS de la PECOS llevan como extensión o nombre secundario la sigla COM.

Cuando a continuación del Prompt del CCP escribimos un comando (por ej.: DIR, BASIC, COPIA) el CCP verifica si el nombre corresponde a un comando interno. De ser así, se lo ejecuta directamente. De no corresponder el comando al repertorio de comandos internos, el CCP busca en el directorio del disco un archivo cuyo nombre primario sea el tipeado por el usuario y cuya extensión sea COM.

Si el CCPO encuentra tal archivo, lo carga en memoria y transfiere el control a éste (BASIC y COPIA son dos ejemplos de comandos externos). Si el CCP no encuentra el comando ni como interno ni como externo (posiblemente porque está en otro disco o se cometió un error de tipeo) emite el mensaje: XXXXXXXX? indicando que desconoce el comando.

El repertorio posible de comandos externos es ilimitado, ya que los programas de aplicación compilados se convierten en nuevos comandos del sistema. Por ejemplo un programa escrito en BASIC compilado para llevar el Libro Diario General de la contabilidad de una empresa puede llamarse DIARIO.COM e invocarse mediante CCP simplemente tipeando DIARIO a continuación del Prompt.

Los comandos transitorios cuando terminan su actividad retransfieren el control al CCP y queda a la espera de un nuevo comando interno o externo.

De la ilimitada gama de comandos posibles vamos a describir en este capítulo los propósitos y forma de uso de aquellos que se encuentran en el disco de distribución que el usuario recibe con el equipo básico.

Si reseteamos la PECOS con el disco de distribución en la disketera, una vez pasado el diálogo inicial (ver guía de operaciones) y teniendo en pantalla el Prompt A> podemos proceder a solicitar el comando DIR. Inmediatamente luego de la actividad de disco observaremos en pantalla el directorio del diskette de distribución. Veremos que éste contiene al menos los siguientes comandos, aunque no necesariamente en el orden en que aquí los enumeramos:

FORMAT.COM

GENSIS.COM

COPIA.COM

GBASIC.COM

BASIC.COM

STAT.COM

COL40.COM

COL40B.COM

COLOR.COM

SUBMIT.COM

COMANDO FORMAT

Este comando permite INICIALIZAR un disco. Inicializar significa grabar en un diskette virgen determinadas marcas especiales que son usadas por el hardware del controlador de diskette para ubicarse sobre la superficie magnética.

Ningún diskette puede utilizarse sin previamente haberlo formateado. Debe tenerse en cuenta que el formateo se hace en general por única vez cuando el diskette es nuevo. Si se formatea un diskette que contiene datos, ESTOS SE BORRAN Y SE PIERDEN DEFINITIVAMENTE SIN POSIBILIDAD DE RECUPERARLOS.

El format se invoca simplemente tipeando FORMAT (return) a continuación del Prompt del CCP y aparece en pantalla un mensaje que dice:

FORMATEAR UN DISCO IMPLICA

DESTRUIR LA INFORMACION QUE

ESTE POSEIA. RETIRE EL DISCO

QUE POSEE ESTE PROGRAMA Y

LUEGO COLOQUE EL DISCO A SER

FORMATEADO

INSERTE EL DISCO EN LA DISKETERA

- PARA COMENZAR DIGITE F

- PARA TERMINAR DIGITE RETURN

Si se digita return (return) se vuelve al CCP y si digitamos F, comienza el proceso de formateado y aparece en pantalla un contador que indica que cara y que pista del diskette se está formateando. Al llegar a CARA 2 PISTA 40, el proceso se detiene y aparece la leyenda

FORMAT TERMINADO

INSERTE EL DISCO EN LA DISKETERA

- PARA COMENZAR DIGITE F

- PARA TERMINAR DIGITE RETURN

Tenemos ahora la posibilidad de formatear más diskettes sin reinvocar al FORMAT o podemos terminar digitando return.

Si por algún motivo no pudiera llevarse a cabo la operación de formateado aparece el mensaje

********** ERROR **********

***** NO PUEDE FORMATEAR *****

Esto se debe, en la mayoría de los casos, a que el diskette a formatear tiene colocado un autoadhesivo de write protect, o sea de inhibición de escritura, obturando una pequeña muesca que tiene sobre el lado izquierdo del envoltorio.

EL COMANDO GENSIS

Un disco formateado está ya en condiciones de ser usado para guardar archivos. De hecho, introduciendo en la disketera un disco en estas condiciones y pidiendo un DIR (luego de hacer ^C por razones que ya hemos visto) obtendremos el mensaje NO FILE, informándonos que efectivamente el diskette está vacío y listo para usar.

Sin embargo, si intentamos resetear la PECOS con este disco veremos que el sistema no llega a poner el Prompt. La razón es que el disco, si bien está formateado, no contiene la copia del sistema operativo en sus pistas de sistema.

La función del GENSIS es depositar una copia del DOS en las pistas de sistema de un diskette previamente formateado, convirtiéndolo de hecho en un disco de sistema apto para resetear la máquina.

El GENSIS se activa como cualquier comando, tipeando su nombre a continuación del Prompt seguido de return.

Aparece entonces la leyenda

DISCO DE ORIGEN

Se debe digitar el nombre de la unidad de disco donde insertaremos el diskette de sistema del cual tomaremos la imagen del DOS. En general la respuesta es "A".

Aparece entonces

INSERTE ORIGINAL EN A

DIGITE <CR>

Donde dice "EN A" si hemos contestado "A" a la pregunta inicial. Una vez insertado el disco del cual queremos leer el DOS, debemos digitar <CR>, o sea return.

En este momento, el GENSIS lee del disco original el DOS. Acto seguido vemos en pantalla

DRIVE DE DESTINO (<CR>=REBOOT)

A esto contestamos con "A" si tenemos una única disketera en la que insertaremos el disco sobre el cual queremos generar el DOS o bien <CR> si deseamos suspender la operación y volver al CCP. Aparece ahora el mensaje

DESTINO EN A

DIGITE <CR>

Una vez que hemos introducido el diskette de destino en la disketera, digitamos <CR> y el GENSIS escribe la imagen del DOS en las pistas de sistema, quedando terminada la operación. Aparece el mensaje

GENSIS TERMINADO

DRIVE DE DESTINO (<CR>=REBOOT)

Estamos ahora en condiciones de repetir la operación de escritura sobre cuantos diskettes queramos o bien digitando <CR> volvemos al CCP, dando por terminado el GENSIS.

El mensaje de error

* ERROR DE ESCRITURA *

en caso de aparecer, obedece a las mismas causas detalladas para el comando FORMAT.

EL COMANDO COPIA

Este comando sirve para copiar archivos de un diskette a otro en un sistema que dispone de una única disketera. Obviamente la única manera de hacer esto es copiar el archivo de origen (o parte de éste si es muy largo) a memoria principal, cambiar el disco de origen por el de destino y copiando entonces el contenido de la memoria al archivo de destino.

El programa COPIA está especialmente optimizado para esta operatoria, emitiendo por pantalla mensajes precisos respecto del cambio de discos. También está optimizado el aprovechamiento de la memoria disponible de manera de mantener en un mínimo la cantidad de veces que se precisa cambiar un diskette por otro.

El programa se invoca por su nombre desde el Prompt del CCP. Una vez que el COPIA está funcionando se puede solicitar la copia de uno o más archivos, especificados mediante una RMA. Si los archivos que satisfacen al RMA son relativamente cortos, el COPIA baja a memoria tantos archivos como se pueda de acuerdo al espacio disponible, a efectos de minimizar la cantidad de cambios de disco.

A medida que progresan la o las copias requeridas el COPIA emite por pantalla el nombre del archivo que en ese momento se está procesando.

Una vez invocado el comando COPIA, la sesión transcurre de la siguiente manera: en primer lugar aparece un asterisco sobre el margen izquierdo. Este es el Prompt del comando COPIA y significa "estoy listo para recibir órdenes". Supongamos que quiero generar una copia del disco de distribución, en tal caso a continuación del prompt escribo

```
*.*
```

que ya sabemos que es una RMA que es satisfecha por todos los archivos del disco. Aparece inmediatamente la leyenda

```
INSERTE DISCO FUENTE
y oprima RETURN
```

Cuando cumplimos con la orden y digitamos RETURN aparecerá en pantalla:

```
Copiando–
COPIA.COM
FORMAT.COM
GENSIS.COM
......
......
```

Hasta que aparece el mensaje

Inserte disco DESTINO

y oprima RETURN

Luego de insertado el disco correspondiente y de haber digitado RETURN (return), se vuelca al nuevo disco los archivos recién leídos y finaliza la operación con el mensaje

Inserte disco SISTEMA

y oprima RETURN

Como se ve, la operatoria es sumamente eficiente y sencilla. Los eventuales mensajes de error se refieren en general a que no existe en el disco de destino espacio suficiente para escribir todo lo que se solicitó.

COMANDO BASIC

Este comando se invoca al igual que los antes mencionados y se trata del intérprete BASIC (ver manual correspondiente).

COMANDO GBASIC

Este comando instala las rutinas de interfase entre el BASIC.COM y los programas gráficos del sistema operativo. A continuación procede a cargar y transferir control al BASIC.COM.

Para poder usar GBASIC, el mismo diskette debe contener el archivo BASIC.COM.

Todo programa que utilice gráficos (ver apéndice sobre gráficos en la PECOS) debe usar forzosamente GBASIC. Aquellos programas que no utilizan gráficos pueden usar GBASIC o BASIC, indistintamente, teniendo en cuenta que GBASIC deja un área de memoria disponible para programas ligeramente menores a BASIC.

COMANDO STAT

Este comando tiene por objeto informar datos estadísticos relativos al disco que se encuentra en la disketera y los archivos que éste contiene.

En su uso más simple, se lo invoca como STAT (return). Esto da como resultado la información por pantalla de la cantidad de kilobytes libres en el disco. O sea es una medida de cuán lleno está el disco. También se nos informa si el disco es R/W (read/write, o sea lee/escribe) o bien R/O (read only, o sea sólo lectura). Recordemos que cambiar de diskette sin hacer ^C convierte el diskette temporariamente en R/O, y que en este caso la información de capacidad libre disponible puede ser errónea.

Otra manera de usar el STAT es invocándolo como:

STAT rma (return)

esta forma genera una lista de todos los archivos que satisfacen la RMA, en orden alfabético e indicando la cantidad de registros (records), kilobytes y extents que cada archivo ocupa.

La cantidad de kilobytes (un kilobyte son 1024 bytes) es el espacio efectivamente ocupado en el disco. En la PECOS, el espacio de disco se va utilizando de a 2 kilobytes, por lo tanto este número será siempre igual o mayor que la octava parte de la cantidad de registros, redondeado hacia arriba al próximo número par de kilobytes.

La cantidad de extents es un dato técnico interno, pero a efectos ilustrativos diremos que representa cuantos renglones del directorio o índice del diskette ocupa un archivo dado. En la PECOS cada extent tiene 32 kbytes de capacidad. Todo archivo de 1 byte a 32767 (32 kbytes) ocupa un solo extent; archivos de 32768 a 65535 bytes ocupan 2 extents, y así sucesivamente.

El directorio de un diskette PECOS tiene capacidad para 64 archivos de un extent, ó 63 archivos de 1 extent y uno de dos extents, o por ejemplo 16 archivos de 4 extents cada uno. La sumatoria de extensiones de un diskette no pueden exceder 64.

La última forma mencionada de uso del STAT también proporciona al final de la lista de archivos la cantidad de kilobytes libres en el diskette.

NOTA: En modo COLOR, debido a que sólo se cuenta con 32 columnas, el display del STAT puede a primera vista aparecer confuso, no obstante la información proporcionada es correcta y sólo se trata de un defecto cosmético debido al escaso número de columnas disponibles en este modo. En modo 40 COLUMNAS, el display no sólo es veraz, sino que es estéticamente correcto.

COMANDO COL40

Este comando pone a la PECOS en el modo 40 COLUMNAS. Este comando es ideal para cuando la pantalla es un televisor o monitor color.
Inicialmente el display es de letras negras sobre fondo cyan (turquesa claro), aunque estos colores pueden ser modificados por el usuario (ver apéndice correspondiente a secuencias de escape en modo 40 columnas).

COMANDO COL40B

Cumple exactamente la misma función que el COL40, pero es ideal para usar con un televisor o monitor monocromo, ya que inicialmente genera letras blancas sobre fondo negro.

COMANDO COLOR

Este comando sirve para pasar de vuelta de modo 40 COLUMNAS a modo COLOR. O sea repone a la PECOS en las condiciones en que arrancó, si es que en algún momento se pasó a 40 columnas. Si no hubo pasaje a 40 columnas, este comando sólo borra la pantalla alfanumérica y la gráfica y vuelve todos los parámetros seleccionables de color a sus condiciones asumidas de arranque.

COMANDO SUBMIT

Este comando es para un uso relativamente sofisticado. Hasta ahora siempre vimos cómo el CCP aceptaba sus comandos del operador, a través de la consola. En realidad, el CCP es capaz de aceptar comandos desde un archivo en disco. Esto permite, en principio, escribir un libreto u hoja de ruta de comandos a ejecutar y el CCP los ejecutará en la secuencia planeada sin la presencia del operador, de forma idéntica a si este último tipeará un nuevo comando cada vez que termina el anterior y reaparece el Prompt del CCP.

Este archivo de "hoja de ruta" debe tener un nombre especial ($$$.SUB) y debe estar escrito de una manera bastante particular.

El comando SUBMIT nos permite escribir nuestra "hoja de ruta" con un editor de textos cualquiera y guardarlo en un archivo con nombre primario a elección y extensión SUB. Luego invocando SUBMIT XXXXXXXX donde las equis representan el nombre elegido para el archivo, el comando se encarga de generar el archivo $$$.SUB en el formato adecuado para el uso del CCP.

Este comando de por sí nos permite facilitar ciertas tareas que implican una lista de comandos que se usan una y otra vez. Pero hay aún más: el SUBMIT permite sustituciones paramétricas en la lista de tareas.

Supongamos que queremos ejecutar una secuencia de tres programas BASIC, distintos entre sí, pero no sabemos a priori el nombre de estos programas. La manera usual de ejecutar un programa BASIC llamado por ejemplo SALDOS es mediante el comando:

```
BASIC SALDOS
```

(ver manual de BASIC). Si con un editor de texto generamos un archivo con el siguiente contenido:

```
BASIC $1
BASIC $2
BASIC $3
```

y lo guardamos en un archivo llamado, por ejemplo, PROC3.SUB podemos luego sencillamente escribir después del Prompt

SUBMIT PROC3 CUENTAS SALDOS CHEQUES

y la PECOS hará exactamente lo mismo que con la siguiente operativa manual por parte de un operador

```
A>BASIC CUENTAS
.....
.....  (se ejecuta el prog CUENTAS)
.....

A>BASIC SALDOS
.....
.....  (se ejecuta el prog SALDOS)
.....

A>BASIC CHEQUES
.....
.....  (se ejecuta el prog CHEQUES)
.....
A>
```

Es interesante notar que donde en el archivo PROC3.SUB escribimos $1, $2 y $3, al generarse el $$$.SUB estos parámetros fueron sustituidos por el primer, segundo y tercer nombre que sigue al nombre del archivo de SUBMIT. Esto se llama sustitución paramétrica. Se pueden incluir todos los parámetros que uno desee $1,$2 . . . . $n, y lo importante es recordar que serán sustituidos en el mismo orden en que aparezcan los nombres en la lista que sigue a la invocación del SUBMIT.

Es de notar que el mismo archivo PROC3.SUB nos permite ejecutar en secuencia cualquier otra serie de tres programas BASIC, por ejemplo DEBITOS, CREDITOS y SALDOS con sólo poner:

SUBMIT PROC3 DEBITOS CREDITOS SALDOS

Si se desea detener la ejecución de un SUBMIT antes que se agote la lista de comandos, se debe pulsar repetidamente la tecla de return en el momento que el CCP toma control momentáneamente entre paso y paso.

CAPITULO 8

MODOS DE FUNCIONAMIENTO

La PECOS contiene un controlador de imagen (procesador de video) sumamente versátil, capaz de trabajar en distintos modos. Cada uno de estos modos ofrece ciertas ventajas y ciertas limitaciones respecto de los otros. En general, la PECOS se usa en uno de dos modos, denominados modo COLOR y modo 40 COLUMNAS. De acuerdo al disco de sistema empleado, la máquina puede arrancar en uno u otro modo. Además, durante el funcionamiento, existen comandos para cambiar de modo sin necesidad de resetear el equipo. Determinados comandos y algunos lenguajes (por ejemplo el MICRO LOGO) precisan modos especiales para funcionar.

MODO COLOR

Este es un modo mixto alfanumérico/gráfico con excelentes posibilidades de manejo y selección de color, tanto para gráficos como para la parte alfanumérica.

La pantalla alfanumérica consta de 768 caracteres dispuestos en 24 filas de 32 columnas con posibilidad de selección de colores, posicionamiento de cursor, tres tipos distintos de borrado, etc. La pantalla gráfica permite efectuar dibujos con una resolución de 256 por 192 puntos (pixels) con posibilidad de ROTULAR (ubicar carteles alfanuméricos) en color superpuestos a los gráficos.

Una pequeña limitación de este modo es que sólo permite un subconjunto de 64 caracteres del código ASCII. Estos son el alfabeto completo en mayúscula y todos los símbolos excluidos la tilde, la barra partida, el acento grave y las llaves de apertura y cierre (~ ñ ` ¨ {)

MODO 40 COLUMNAS

En este modo se dispone de una única pantalla de 960 caracteres dispuestos en 24 filas de 40 columnas. La única posibilidad de color es la de seleccionar a voluntad el color de los caracteres y del fondo de éstos. La ventaja fundamental reside en que para información textual es más cómodo trabajar con una mayor cantidad de columnas, y el conjunto de caracteres dibujables en pantalla incluye la totalidad del código ASCII, o sea letras mayúsculas, minúsculas, símbolos y caracteres especiales.